



# References

1	CAN Specification Version 2 (1991), Robert Bosch GmbH
2	ISO11898-1 Controller Area Network - Data link layer and physical signalling.
3	ISO11898-2 Controller Area Network - High-speed medium access unit.

# Glossary

API	Application Programming Interface
CAN	Controller Area Network
CRC	Cyclic Redundancy Check
DLC	Data Length Code
EMC	ElectroMagnetic Compatibility
EU	European Union
ISO	International Organization for Standardization
LED	Light Emitting Diode
PC	Personal Computer
RAM	Random Access Memory
ROM	Read Only Memory
SDK	Software Development Kit
USB	Universal Serial Bus

# System Requirements

CANdo application minimum PC system requirements -

- Linux (kernel v2.6 or newer) operating system
- Pentium III 500MHz or equivalent
- Display with a resolution of 1024x768 pixels & 256 colours
- USB port (version 1.1, 2.0 or 3.0 compliant).
- CD ROM drive
- 256M bytes of RAM
- 10M bytes of free disk space

## New

'New', clears the current configuration.

The 'CAN View' mode, 'CAN Transmit' list, 'CAN Transmit Editor', & 'CAN Setup' are all cleared to the default configuration. The configuration file name in the title bar is cleared & the user is prompted to check the 'CAN Setup' for the new configuration.

# Load

'Load', loads a previously saved configuration from a file on the PC.

The 'CAN View' mode, 'CAN Transmit' list, 'CAN Transmit Editor' & 'CAN Setup' are all retrieved from the selected configuration file.

## Save/Save As

'Save', saves the current configuration to a user specified file on the PC.

In the file dialog box that appears on pressing 'Save', type a file name for the current configuration to be saved (the file extension is automatically added). The file name of the saved configuration appears in the title bar. Pressing 'Save' again saves any changes to the specified file without bringing up the file dialog box again.

Pressing 'Save As' allows a new configuration file name to be specified for the current configuration.

# Exit CANdo

'Exit', closes the **CANdo** application & the connection to **CANdo**.

If changes have been made to the 'CAN Transmit' list, 'CAN Transmit Editor' or 'CAN Setup', then the user is prompted to save the configuration before exiting.

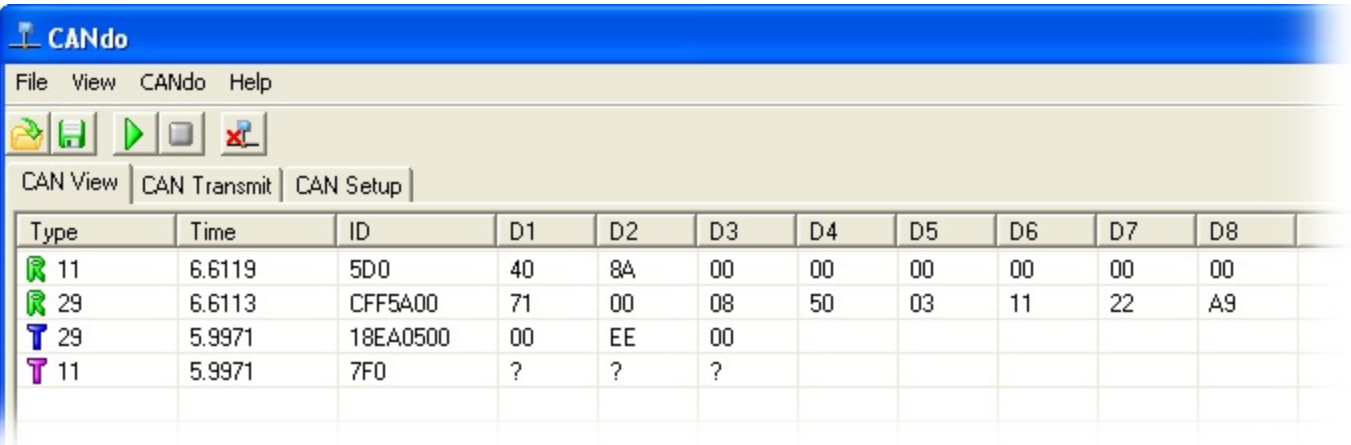


## Clear

Clear, erases all the current messages displayed in the 'CAN View' window.

# Fixed

In 'Fixed' view mode, the received & transmitted CAN messages are listed in the 'CAN View' window in fixed positions. Any message with the same 'Type' & ID as one already displayed automatically overwrites the currently displayed message. A new message with no equivalent 'Type' & 'ID' as one already in the list, is appended to the end.



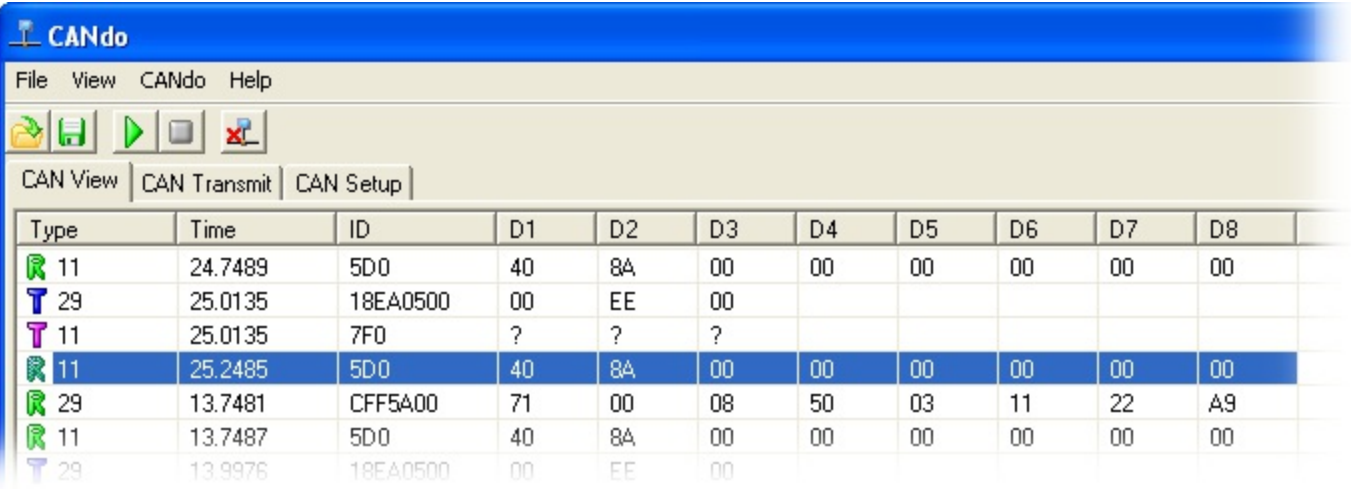
The screenshot shows the CANdo software interface. The title bar is blue with the text 'CANdo'. Below it is a menu bar with 'File', 'View', 'CANdo', and 'Help'. A toolbar contains icons for opening a file, saving, running, and other functions. Below the toolbar are three tabs: 'CAN View' (selected), 'CAN Transmit', and 'CAN Setup'. The 'CAN View' tab displays a table of CAN messages. The table has columns for Type, Time, ID, and data bytes D1 through D8. The data is as follows:

Type	Time	ID	D1	D2	D3	D4	D5	D6	D7	D8
R 11	6.6119	5D0	40	8A	00	00	00	00	00	00
R 29	6.6113	CFF5A00	71	00	08	50	03	11	22	A9
T 29	5.9971	18EA0500	00	EE	00					
T 11	5.9971	7F0	?	?	?					

A context sensitive menu appears if the right mouse button is clicked while hovering over the 'CAN View' list. This menu allows either 'Fixed' or 'Cyclic' view mode to be selected.

# Cyclic

In 'Cyclic' view mode, the CAN messages received & transmitted are listed in the 'CAN View' in chronological order, with new messages being added to the end of the list.



Type	Time	ID	D1	D2	D3	D4	D5	D6	D7	D8
R 11	24.7489	5D0	40	8A	00	00	00	00	00	00
T 29	25.0135	18EA0500	00	EE	00					
T 11	25.0135	7F0	?	?	?					
R 11	25.2485	5D0	40	8A	00	00	00	00	00	00
R 29	13.7481	CFF5A00	71	00	08	50	03	11	22	A9
R 11	13.7487	5D0	40	8A	00	00	00	00	00	00
T 29	13.9976	18EA0500	00	EE	00					

After 500 messages have been added to the list, the list cycles back to the start, with the newest message replacing the oldest one.

If the total (received & transmitted) message rate is greater than 250 messages per second, the view mode automatically reverts back to 'Fixed' view mode.

A context sensitive menu appears if the right mouse button is clicked while hovering over the 'CAN View' list. This menu allows either 'Fixed' or 'Cyclic' view mode to be selected.

# Start CANdo

'Start CANdo', initialises the operating mode of the CANdo unit & enables the reception & transmission of CAN messages. The CAN message timestamp is reset to 0s.

The 'Esc' key is a short cut that toggles between 'Start CANdo' & 'Stop CANdo'.

# Stop CANdo

'Stop **CANdo**', disables the reception & transmission of CAN messages.

The 'Esc' key is a short cut that toggles between 'Start **CANdo**' & 'Stop **CANdo**'.

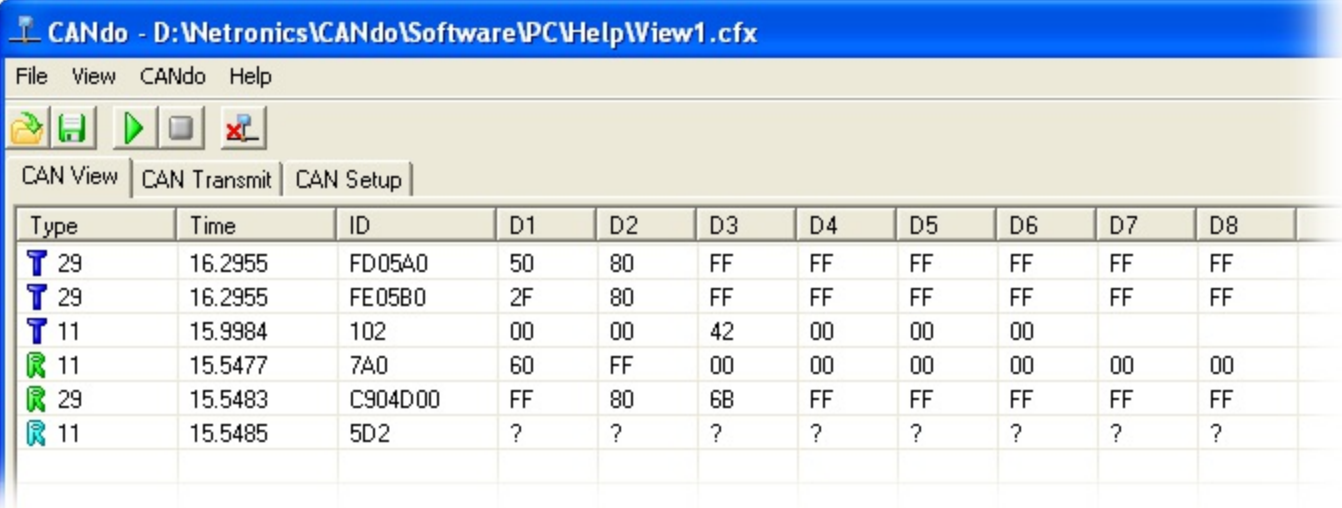
'CANdo Help', invokes this built in help.

# About CANdo

'About **CANdo**', brings up a dialog box displaying the name & version number of the **CANdo** application.

# Overview

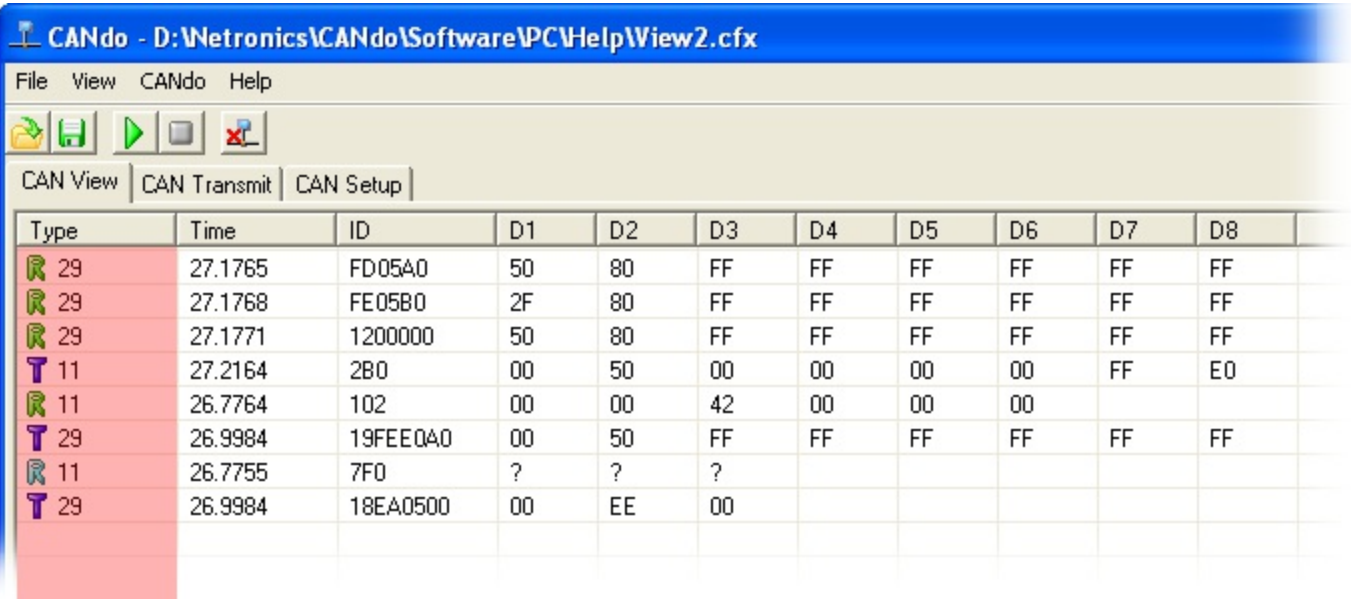
The 'CAN View' is the main CAN message display window. All CAN messages transmitted & received on the CAN bus are displayed within this window. Error frames & CAN bus state changes are also displayed as & when they occur.





# Type

The 'Type' Column in the 'CAN View' window displays the type of message on each row in the list, using an icon & either a numerical value or some text. The type identifies the message as either receive, transmit or status. The numbers '11' & '29' appended to the type icon in the case of a receive or a transmit, identify the message as either a CAN 2.0A (11 bit standard arbitrator) or CAN 2.0B (29 bit extended arbitrator) message, respectively.



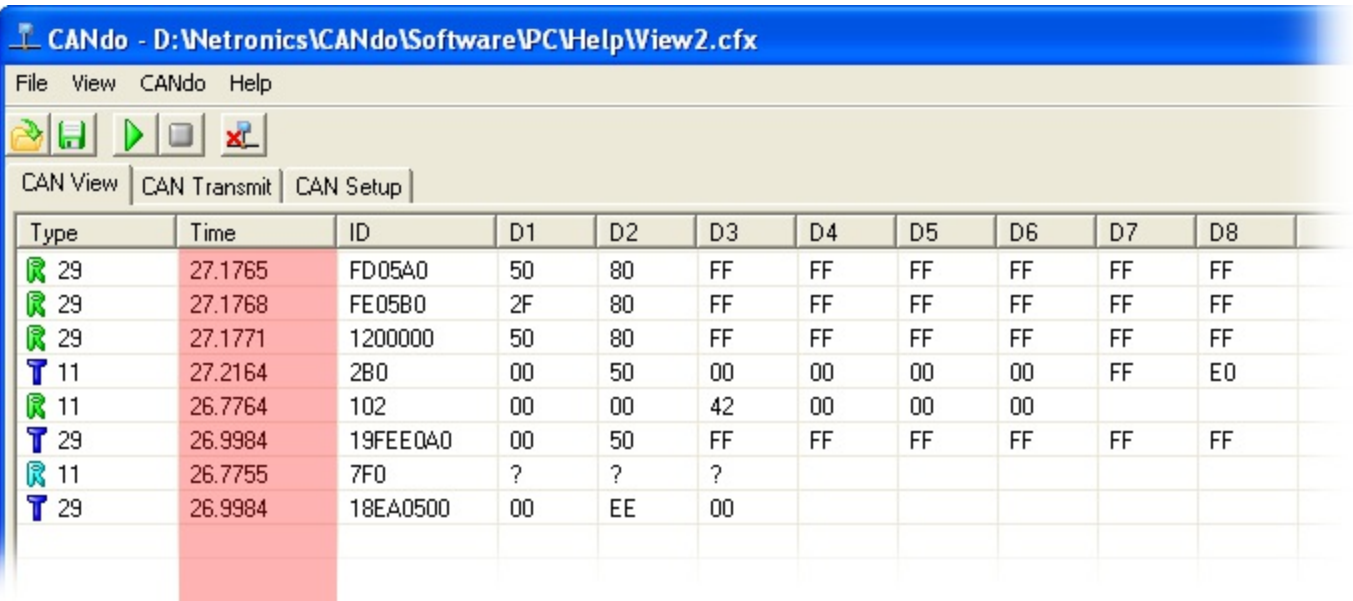
Type	Time	ID	D1	D2	D3	D4	D5	D6	D7	D8
R 29	27.1765	FD05A0	50	80	FF	FF	FF	FF	FF	FF
R 29	27.1768	FE05B0	2F	80	FF	FF	FF	FF	FF	FF
R 29	27.1771	1200000	50	80	FF	FF	FF	FF	FF	FF
T 11	27.2164	2B0	00	50	00	00	00	00	FF	E0
R 11	26.7764	102	00	00	42	00	00	00		
T 29	26.9984	19FEE0A0	00	50	FF	FF	FF	FF	FF	FF
R 11	26.7755	7F0	?	?	?					
T 29	26.9984	18EA0500	00	EE	00					

The table below details the possible message types.

Icon	Type
R 11	Receive message with 11 bit arbitrator
R 29	Receive message with 29 bit arbitrator
R 11	Remote receive message with 11 bit arbitrator
R 29	Remote receive message with 29 bit arbitrator
T 11	Transmit message with 11 bit arbitrator
T 29	Transmit message with 29 bit arbitrator
T 11	Remote transmit message with 11 bit arbitrator
T 29	Remote transmit message with 29 bit arbitrator
B Active	Less than 128 receive/transmit errors
B Passive	More than 128 receive/transmit errors
B Off	More than 255 transmit errors
E Error	Error frame detected

# Time

The 'Time' column in the 'CAN View' window displays a timestamp for each message. The timestamp is the arrival time of the message in seconds elapsed since **CANdo** started.



The screenshot shows the CANdo software interface. The title bar reads "CANdo - D:\Netronics\CANdo\Software\PCVHelpView2.cfx". The menu bar includes "File", "View", "CANdo", and "Help". Below the menu bar is a toolbar with icons for opening files, saving, running, and other functions. The main window has three tabs: "CAN View", "CAN Transmit", and "CAN Setup". The "CAN View" tab is active, displaying a table of CAN messages. The table has columns for Type, Time, ID, and data bytes D1 through D8. The "Time" column is highlighted in red. The messages are as follows:

Type	Time	ID	D1	D2	D3	D4	D5	D6	D7	D8
R 29	27.1765	FD05A0	50	80	FF	FF	FF	FF	FF	FF
R 29	27.1768	FE05B0	2F	80	FF	FF	FF	FF	FF	FF
R 29	27.1771	1200000	50	80	FF	FF	FF	FF	FF	FF
T 11	27.2164	2B0	00	50	00	00	00	00	FF	E0
R 11	26.7764	102	00	00	42	00	00	00		
T 29	26.9984	19FEE0A0	00	50	FF	FF	FF	FF	FF	FF
T 11	26.7755	7F0	?	?	?					
T 29	26.9984	18EA0500	00	EE	00					

The timestamp is displayed to a resolution of 100us & is accurate to within +/- (20ms + 100ppm). The timestamp wraps back to 0s on reaching 109952s, approximately 30.5 hours.

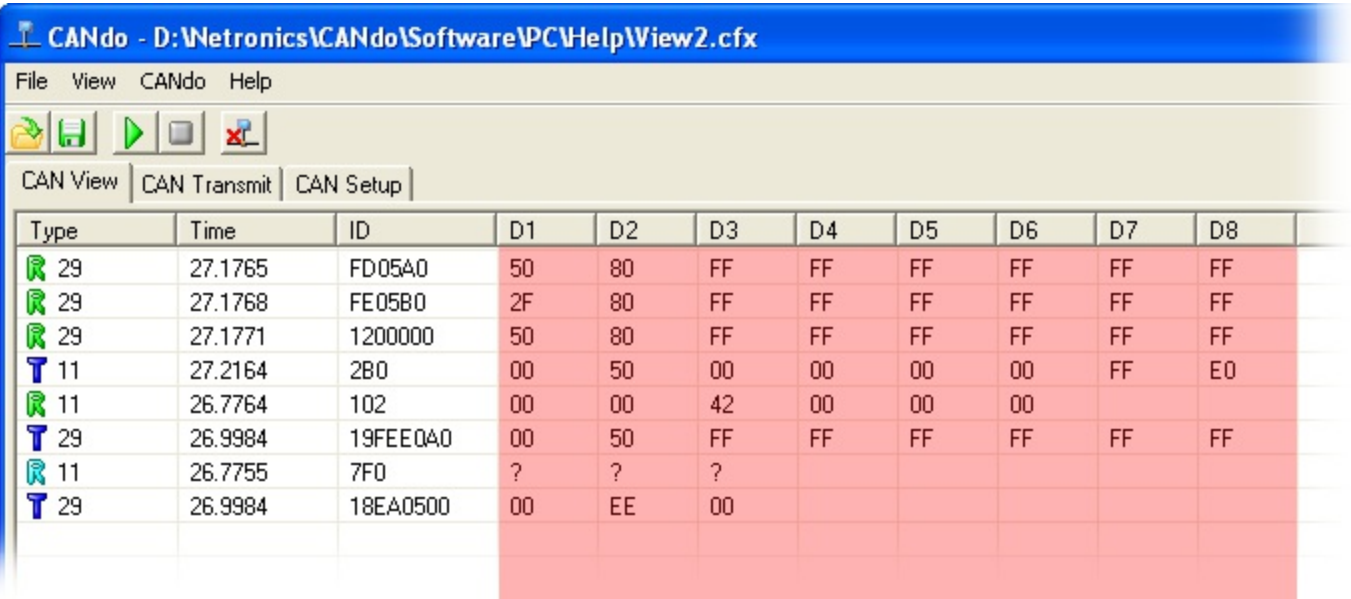
# ID

The 'ID' column in the 'CAN View' window displays the 11 or 29 bit message arbitrator in hexadecimal (radix 16) format.

11 bit arbitrators (CAN 2.0A) lie in the range 000 to 7FF & 29 bit arbitrators (CAN 2.0B) in the range 00000000 to 1FFFFFFF.

# Data

The 'Data' column in the 'CAN View' window displays the data field associated with each CAN message. The data field may occupy from 0 to 8 bytes, these are displayed in columns D1 to D8. If the number of bytes in the data field is less than 8 then the remainder are filled with blanks. The data is displayed in hexadecimal format & each byte lies in the range 00 to FF.

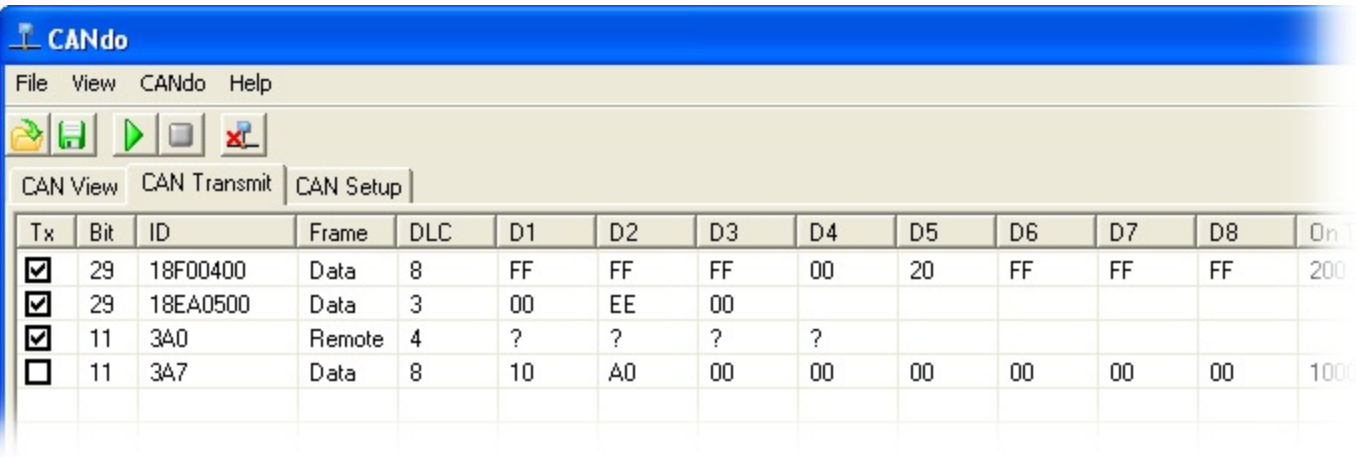


Type	Time	ID	D1	D2	D3	D4	D5	D6	D7	D8
R 29	27.1765	FD05A0	50	80	FF	FF	FF	FF	FF	FF
R 29	27.1768	FE05B0	2F	80	FF	FF	FF	FF	FF	FF
R 29	27.1771	1200000	50	80	FF	FF	FF	FF	FF	FF
T 11	27.2164	2B0	00	50	00	00	00	00	FF	E0
R 11	26.7764	102	00	00	42	00	00	00		
T 29	26.9984	19FEE0A0	00	50	FF	FF	FF	FF	FF	FF
R 11	26.7755	7F0	?	?	?					
T 29	26.9984	18EA0500	00	EE	00					

Remote receive/transmit messages are requests for data & so the data fields are blank. The requested number of data bytes is indicated by the corresponding number of '?'s in the D1 to D8 columns.

# Overview

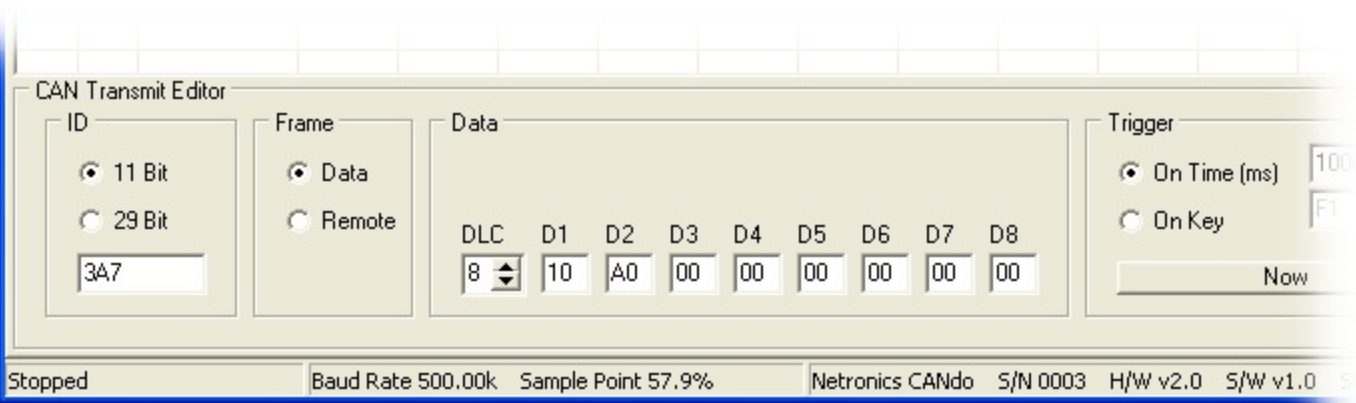
CANdo transmits messages on the CAN bus as specified by the user in the 'CAN Transmit' list, on the 'CAN Transmit' page.



The screenshot shows the CANdo software window with the 'CAN Transmit' tab selected. It displays a table of transmit messages with columns for Tx, Bit, ID, Frame, DLC, and data bytes D1 through D8, plus an 'On' column. The first three rows are checked, and the fourth is unchecked.

Tx	Bit	ID	Frame	DLC	D1	D2	D3	D4	D5	D6	D7	D8	On
<input checked="" type="checkbox"/>	29	18F00400	Data	8	FF	FF	FF	00	20	FF	FF	FF	200
<input checked="" type="checkbox"/>	29	18EA0500	Data	3	00	EE	00						
<input checked="" type="checkbox"/>	11	3A0	Remote	4	?	?	?	?					
<input type="checkbox"/>	11	3A7	Data	8	10	A0	00	00	00	00	00	00	1000

Transmit messages are created & modified using the 'CAN Transmit Editor'.



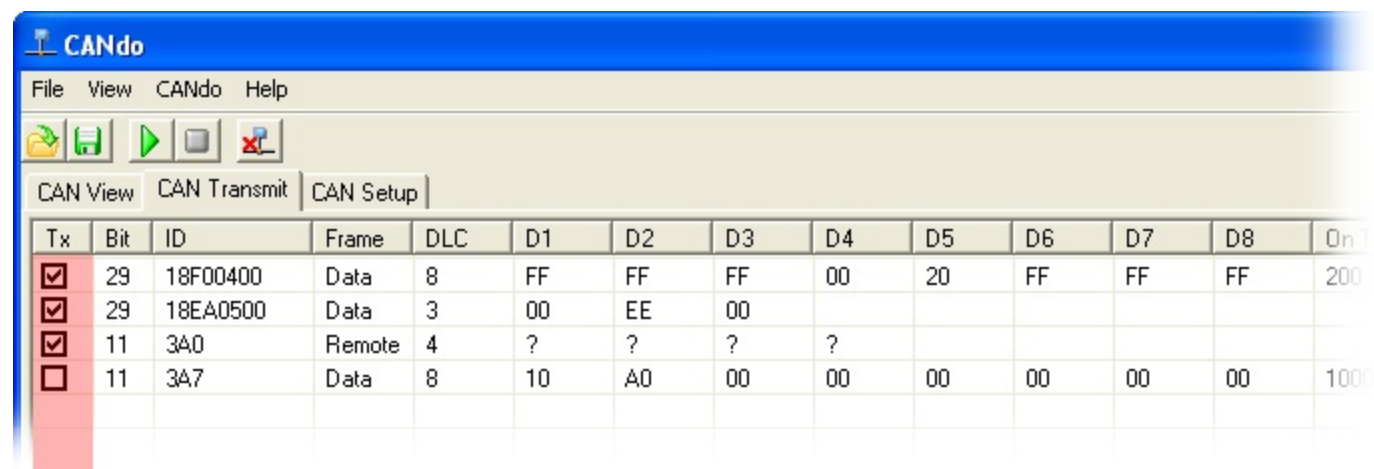
The screenshot shows the 'CAN Transmit Editor' dialog box. It has four main sections: ID, Frame, Data, and Trigger. The ID section has radio buttons for '11 Bit' and '29 Bit', with a text field containing '3A7'. The Frame section has radio buttons for 'Data' and 'Remote'. The Data section has a table for DLC and data bytes D1 through D8. The Trigger section has radio buttons for 'On Time (ms)' and 'On Key', with a 'Now' button.

DLC	D1	D2	D3	D4	D5	D6	D7	D8
8	10	A0	00	00	00	00	00	00

Up to 50 transmit messages may be specified in this way.

## Tx

The 'Tx' box in the 'CAN Transmit' list allows the transmission of individual messages to be enabled/disabled. Any message with the box ticked is enabled for transmission.

[illegible]

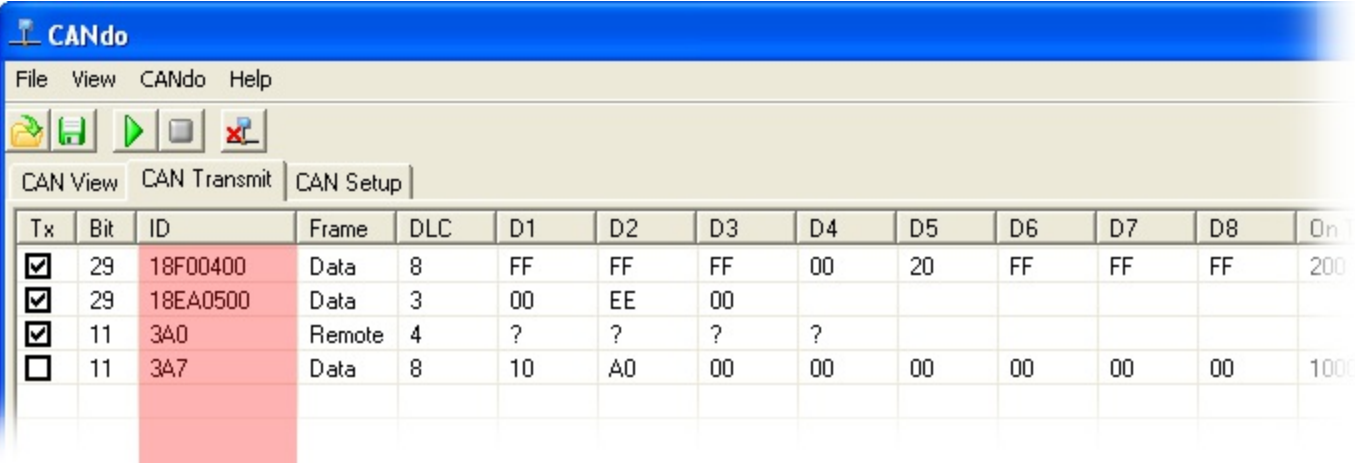
# Bit

The 'Bit' value in the 'CAN Transmit' list identifies the number of bits in the arbitrator of the message, either 11 or 29.

CANdo													
File View CANdo Help													
CAN View CAN Transmit CAN Setup													
Tx	Bit	ID	Frame	DLC	D1	D2	D3	D4	D5	D6	D7	D8	On
<input checked="" type="checkbox"/>	29	18F00400	Data	8	FF	FF	FF	00	20	FF	FF	FF	200
<input checked="" type="checkbox"/>	29	18EA0500	Data	3	00	EE	00						
<input checked="" type="checkbox"/>	11	3A0	Remote	4	?	?	?	?					
<input type="checkbox"/>	11	3A7	Data	8	10	A0	00	00	00	00	00	00	1000

# ID

The 'ID' is the message arbitrator, displayed in hexadecimal format.



The screenshot shows the CANdo software interface. At the top is a blue title bar with the text 'CANdo'. Below it is a menu bar with 'File', 'View', 'CANdo', and 'Help'. Under the menu bar is a toolbar with icons for opening a file, saving, running, and other functions. Below the toolbar are three tabs: 'CAN View', 'CAN Transmit', and 'CAN Setup'. The 'CAN View' tab is active, displaying a table of CAN messages. The table has columns for 'Tx', 'Bit', 'ID', 'Frame', 'DLC', 'D1', 'D2', 'D3', 'D4', 'D5', 'D6', 'D7', 'D8', and 'On'. The 'ID' column is highlighted in red. The table contains four rows of data, with the first three rows having checkboxes in the 'Tx' column.

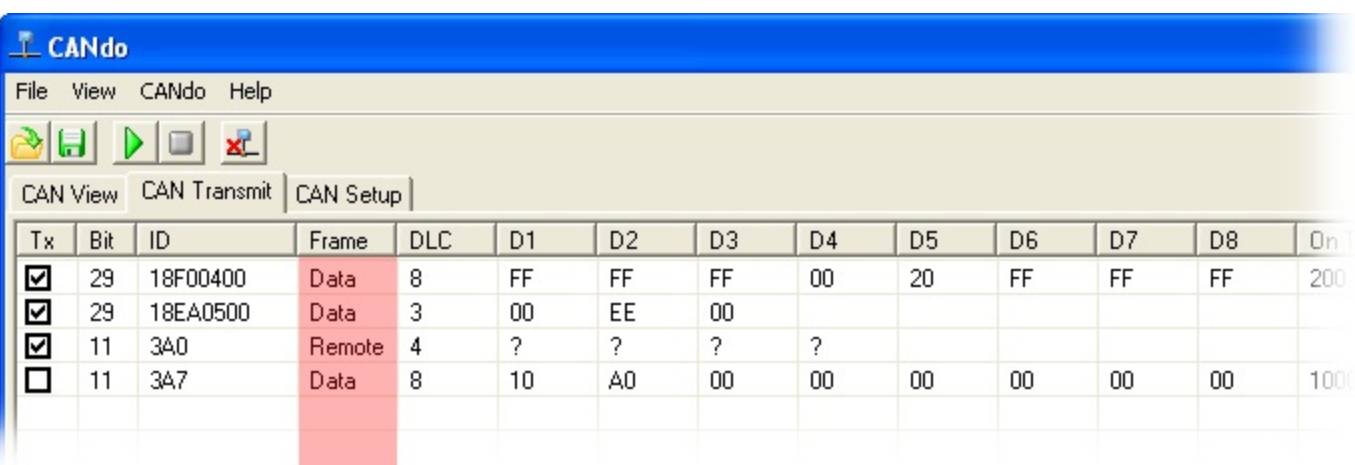
Tx	Bit	ID	Frame	DLC	D1	D2	D3	D4	D5	D6	D7	D8	On
<input checked="" type="checkbox"/>	29	18F00400	Data	8	FF	FF	FF	00	20	FF	FF	FF	200
<input checked="" type="checkbox"/>	29	18EA0500	Data	3	00	EE	00						
<input checked="" type="checkbox"/>	11	3A0	Remote	4	?	?	?	?					
<input type="checkbox"/>	11	3A7	Data	8	10	A0	00	00	00	00	00	00	1000

11 bit arbitrators (CAN 2.0A) lie in the range 000 to 7FF & 29 bit arbitrators (CAN 2.0B) in the range 00000000 to 1FFFFFFF.



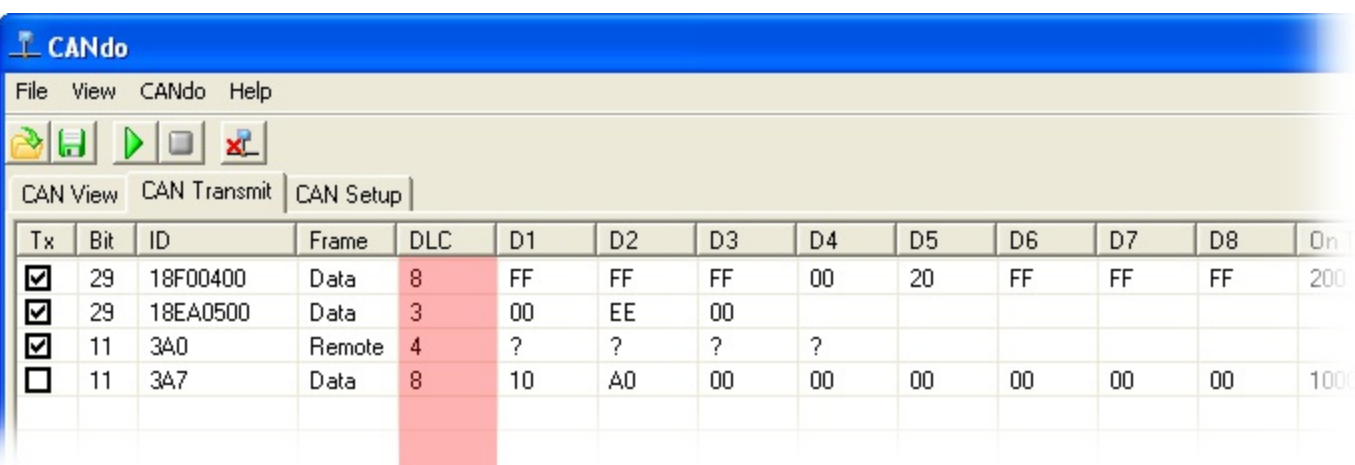
# Frame

The 'Frame' is displayed as either 'Data' for a standard frame containing data or 'Remote' for a remote frame requesting data.



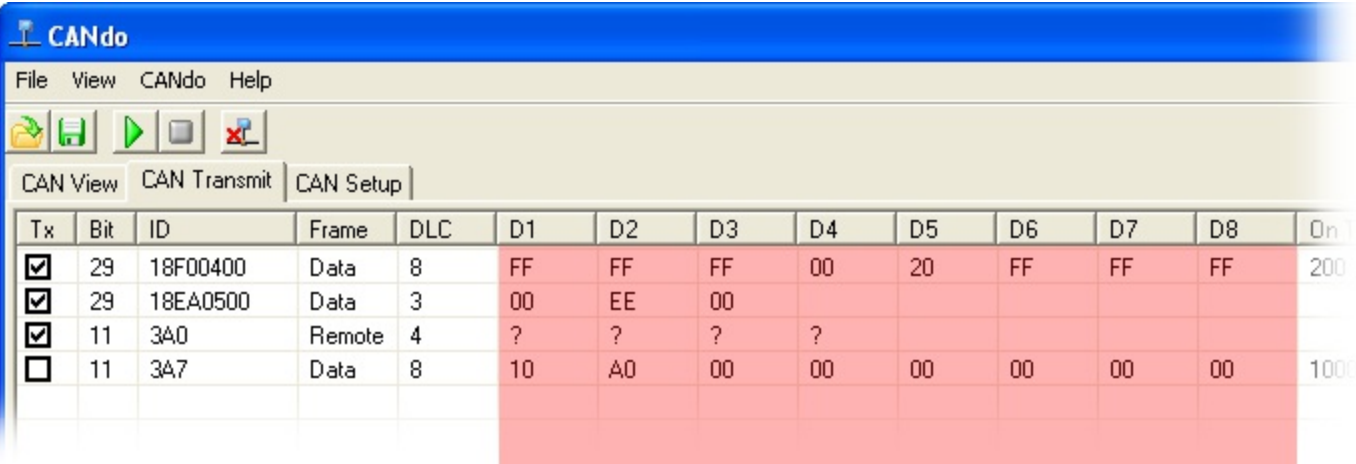
# DLC

The 'DLC' or Data Length Code, is the length of the data field in the message in bytes, ranging from 0 to 8.



# Data

The 'Data' field contains from 0 to 8 bytes of data for a data frame message. For a remote frame message, the 'Data' field bytes are padded with a '?' to indicate the number of bytes requested by the message.



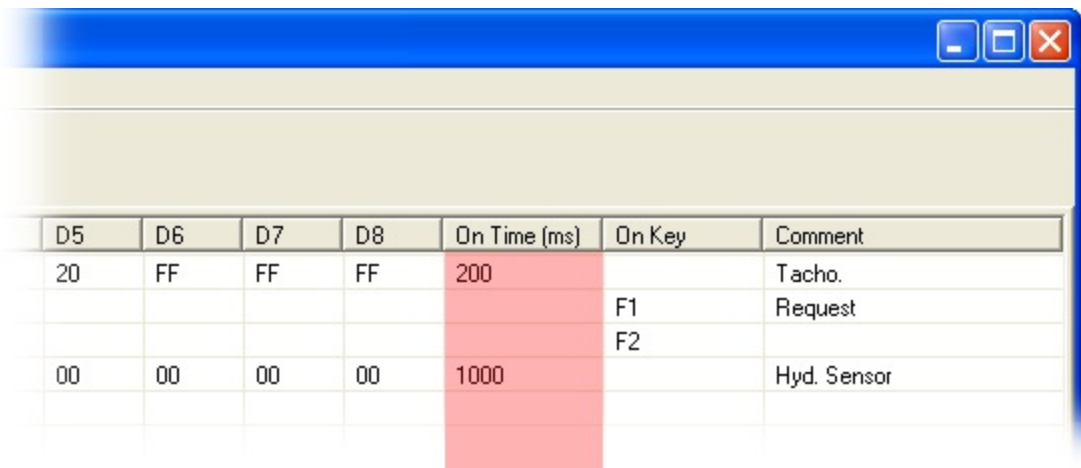
The screenshot shows the CANdo software interface. At the top is a blue title bar with the text 'CANdo'. Below it is a menu bar with 'File', 'View', 'CANdo', and 'Help'. Under the menu bar is a toolbar with icons for opening a file, saving, running, and other functions. Below the toolbar are three tabs: 'CAN View', 'CAN Transmit', and 'CAN Setup'. The 'CAN View' tab is active, displaying a table of CAN bus messages. The table has columns for 'Tx', 'Bit', 'ID', 'Frame', 'DLC', 'D1', 'D2', 'D3', 'D4', 'D5', 'D6', 'D7', 'D8', and 'On'. The data rows are highlighted in red. The first row shows a message with ID 18F00400, Frame 'Data', DLC 8, and data bytes FF, FF, FF, 00, 20, FF, FF, FF. The second row shows a message with ID 18EA0500, Frame 'Data', DLC 3, and data bytes 00, EE, 00. The third row shows a message with ID 3A0, Frame 'Remote', DLC 4, and data bytes ?, ?, ?, ?. The fourth row shows a message with ID 3A7, Frame 'Data', DLC 8, and data bytes 10, A0, 00, 00, 00, 00, 00, 00.

Tx	Bit	ID	Frame	DLC	D1	D2	D3	D4	D5	D6	D7	D8	On
<input checked="" type="checkbox"/>	29	18F00400	Data	8	FF	FF	FF	00	20	FF	FF	FF	200
<input checked="" type="checkbox"/>	29	18EA0500	Data	3	00	EE	00						
<input checked="" type="checkbox"/>	11	3A0	Remote	4	?	?	?	?					
<input type="checkbox"/>	11	3A7	Data	8	10	A0	00	00	00	00	00	00	1000

The data bytes D1 to D8 are displayed in hexadecimal format with a range from 00 to FF.

# OnTime

Transmit messages in the 'CAN Transmit' list with an entry in the 'OnTime' column are cyclic messages that repeat continuously (when enabled), at the rate specified by the 'OnTime' parameter.



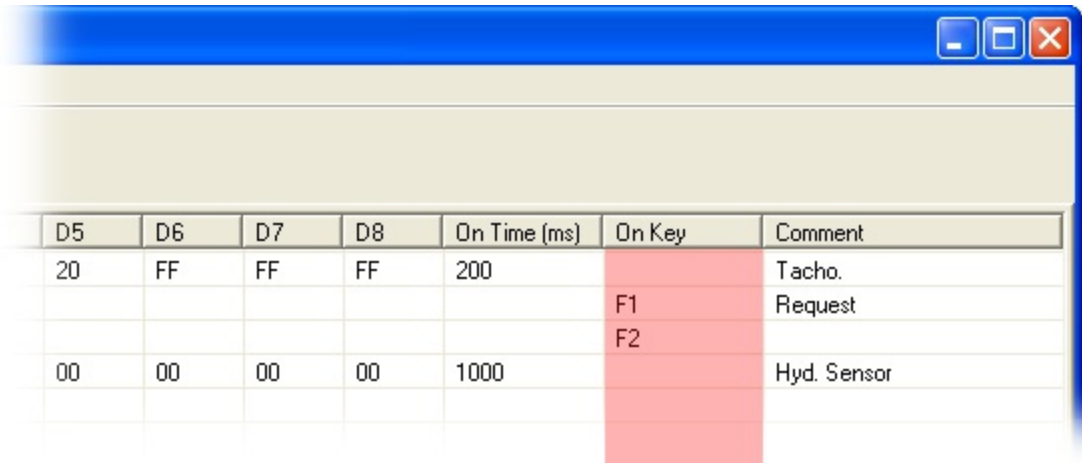
The screenshot shows a software window with a blue title bar and standard Windows window controls (minimize, maximize, close). The window contains a table with the following data:

D5	D6	D7	D8	On Time (ms)	On Key	Comment
20	FF	FF	FF	200		Tacho.
					F1	Request
					F2	
00	00	00	00	1000		Hyd. Sensor

The 'OnTime' repeat value is displayed in milliseconds.

# OnKey

Transmit messages in the 'CAN Transmit' list with an entry in the 'OnKey' column are transmitted (when enabled) in response to pressing the indicated key.

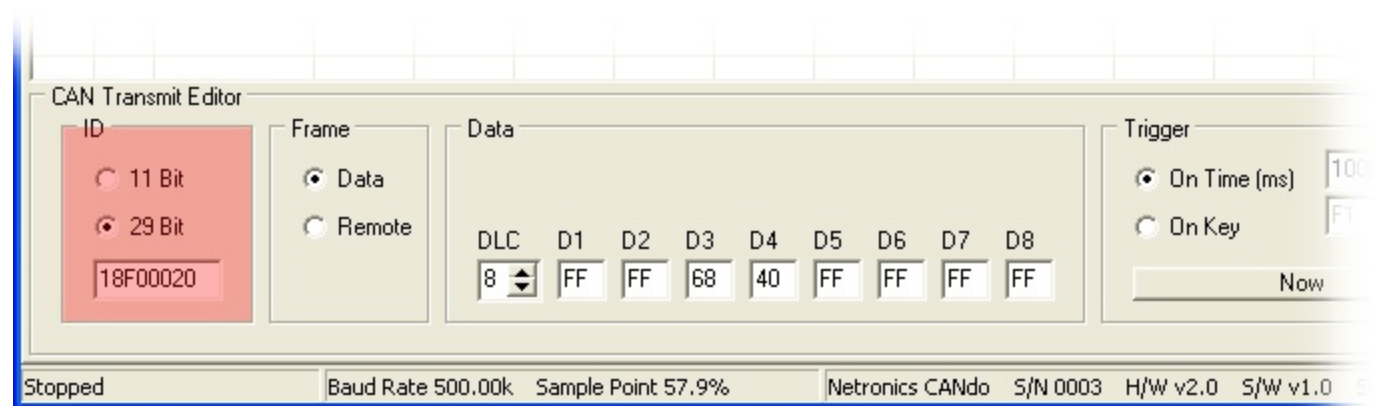


D5	D6	D7	D8	On Time (ms)	On Key	Comment
20	FF	FF	FF	200		Tacho.
					F1	Request
					F2	
00	00	00	00	1000		Hyd. Sensor

Transmit messages are either triggered 'OnTime' or 'OnKey', but not both.

# ID

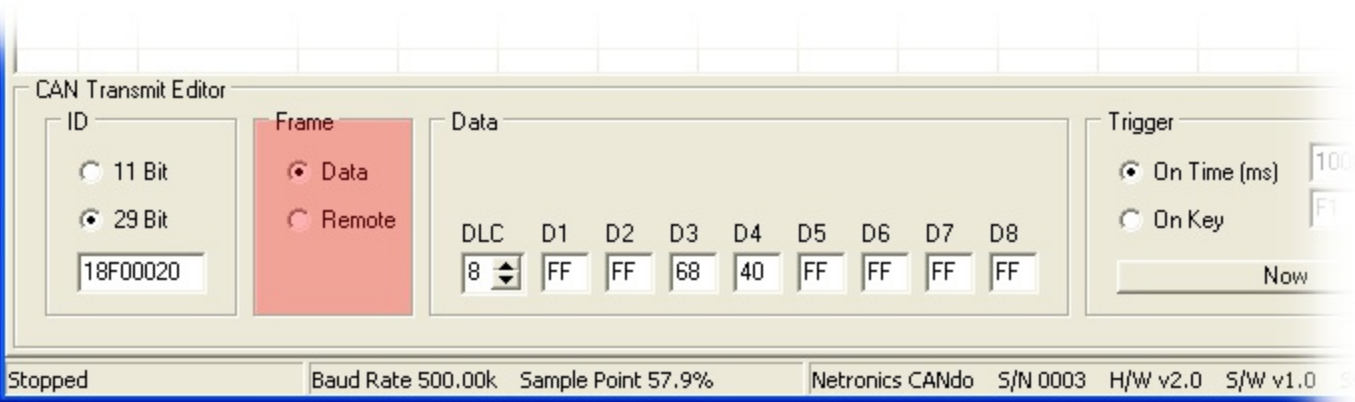
The 'ID' box in the 'CAN Transmit Editor' is used to specify the length & value of the transmit message arbitrator. The arbitrator value is entered in hexadecimal format.



11 bit arbitrators (CAN 2.0A) lie in the range 000 to 7FF & 29 bit arbitrators (CAN 2.0B) in the range 00000000 to 1FFFFFFF.

# Frame

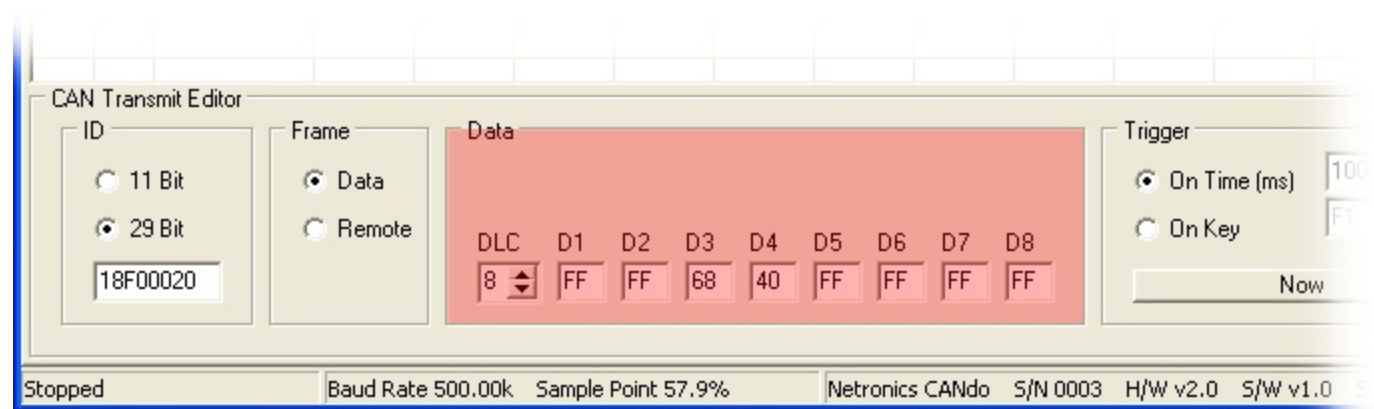
The 'Frame' box in the 'CAN Transmit Editor' is used to specify the transmit message as either a 'Data' frame or a 'Remote' frame.



'Remote' frames are a request for data & therefore do not contain any data fields.

# Data

The 'Data' box in the 'CAN Transmit Editor' is used to specify the data field of the transmit message. The 'DLC' (Data Length Code) specifies the number of bytes (0 to 8) in the data field. The data is entered in hexadecimal format & each byte lies in the range 00 to FF.

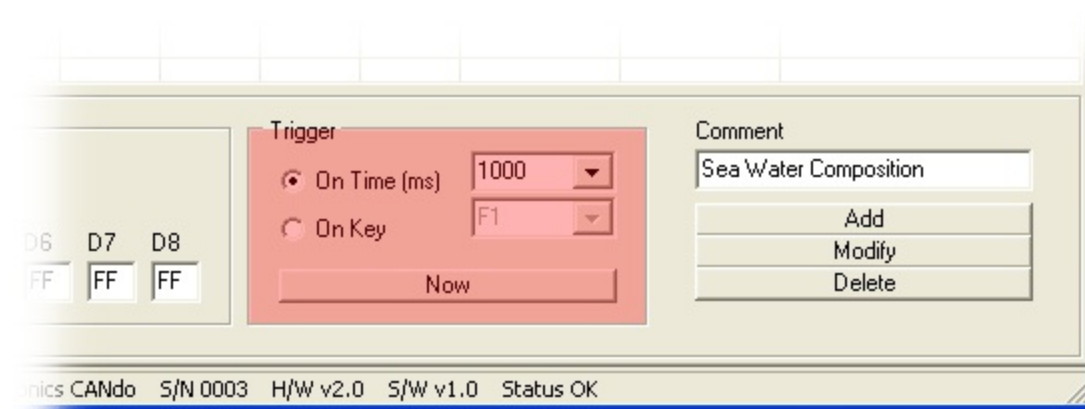


'Remote' frames do not contain a data field, but do contain a DLC. For 'Remote' frames the DLC specifies the number of bytes of data requested by the message.



# Trigger

The 'Trigger' box in the 'CAN Transmit Editor' is used to specify the trigger method used to send the message. The message may either be triggered on a periodic basis ('On Time') or in response to a key press ('On Key').

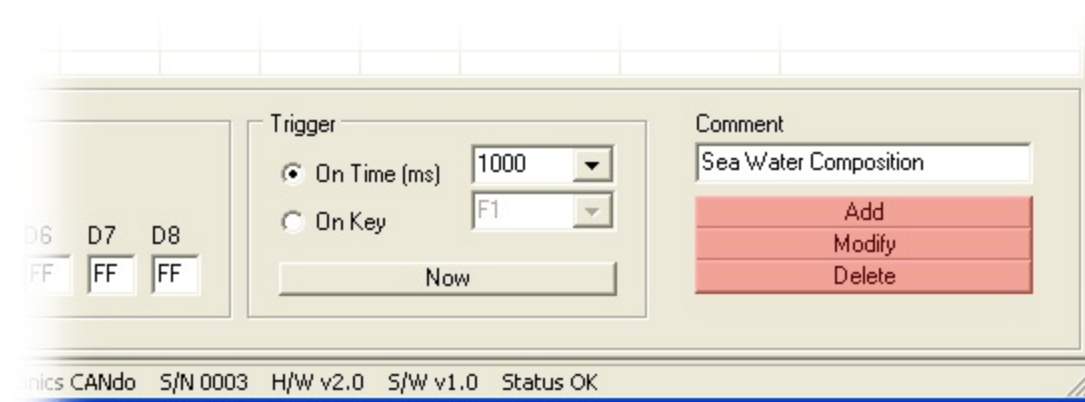


While **CANdo** is running, the current message displayed in the 'CAN Transmit Editor' may be sent immediately by clicking the 'Now' button.

A context sensitive menu appears if the right mouse button is clicked while hovering over the 'CAN Transmit' list. This menu allows a message selected from the list to be sent immediately by clicking on the 'Now' menu item.

# Add/Modify/Delete

The 'Add', 'Modify' & 'Delete' buttons in the 'CAN Transmit Editor' are used to either add, modify or delete a message in the 'CAN Transmit' list using the settings in the 'CAN Transmit Editor'.



The 'Add' button appends the message in the 'CAN Transmit Editor' to the end of the 'CAN Transmit' list. If any of the values are out of range then a warning is given & the message is not added. A maximum of 50 transmit messages may be added.

The 'Modify' & 'Delete' buttons modify & delete respectively, the message selected in the 'CAN Transmit' list. If no message is selected then a warning is given.

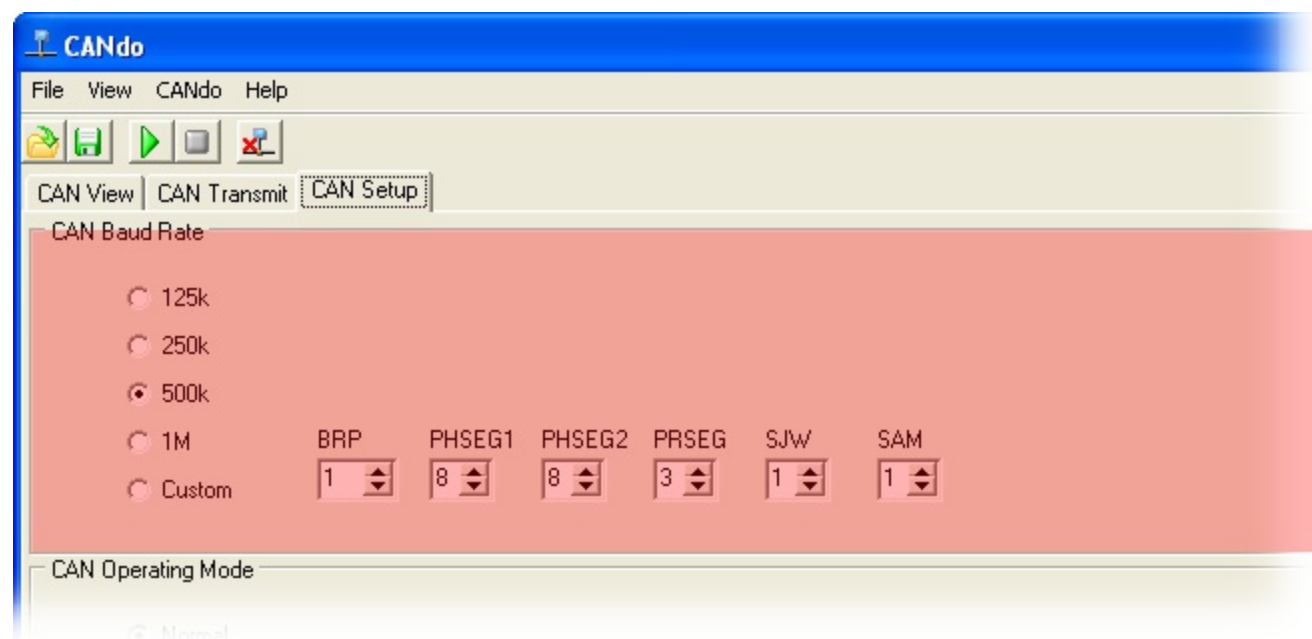
The 'Add' & 'Delete' buttons are disabled while [CANdo](#) is running.

# CAN Baud Rate

The 'CAN Baud Rate' specifies the bit rate on the CAN bus. All nodes connected to a CAN bus must operate at the same baud rate to within +/- 1.7%. The **CANdo** baud rate may be set to any value in the range 6.25k to 1M baud. The currently specified baud rate is automatically calculated & displayed in the status bar.

The baud rate is set by adjusting the following parameters -

- BRP - Baud Rate Prescaler
- PHSEG1 - Phase Segment 1
- PHSEG2 - Phase Segment 2
- PRSEG - Propagation Segment



The baud rate is calculated by the following equation -  
$$20000000 / 2 * BRP * (1 + PHSEG1 + PHSEG2 + PRSEG)$$

For example -  
$$20000000 / 2 * 2 * (1 + 8 + 8 + 3) = 250k \text{ baud}$$

Baud rates 125k, 250k, 500k & 1M are predefined, other baud rates are custom programmable.

The baud rate cannot be changed while **CANdo** is running.

---

The sampling point of a bit on the bus is defined by the following equation -  
$$(1 + PHSEG1 + PRSEG) * 100 / (1 + PHSEG1 + PHSEG2 + PRSEG)$$

For example -

$$(1 + 8 + 3) * 100 / (1 + 8 + 8 + 3) = 60.0\%$$

---

The SJW (Synchronisation Jump Width) specifies the extent to which the PHSEG1 & 2 times may be adjusted by **CANdo** in order to synchronise the bit sampling to the received signal edges on the bus.

---

The SAM (SAMple) value specifies the number of times each bit is sampled. This is usually set to 1, except in noisy environments.

---

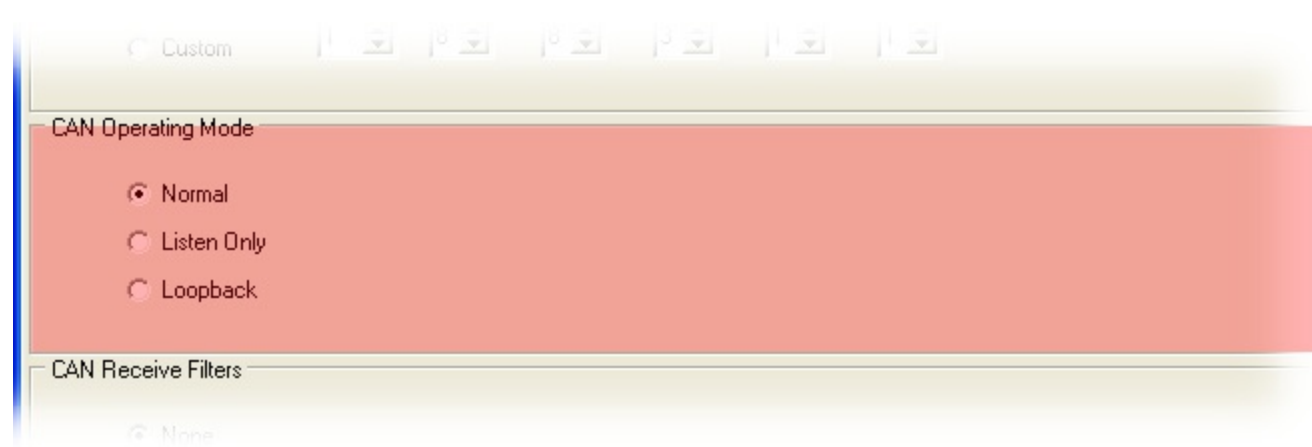
There are some rules that apply to setting the above values as they are to some extent interdependent -

- $\text{PHSEG1} + \text{PRSEG} \geq \text{PHSEG2}$
- $\text{PHSEG1} + \text{PHSEG2} + \text{PRSEG} \geq 7$
- $\text{PHSEG2} \geq \text{SJW}$

# CAN Operating Mode

CANdo may operate in one of three modes -

- Normal
- Listen Only
- Loopback



'Normal' mode, allows full transmission & reception of CAN messages & is the normal mode of operation.

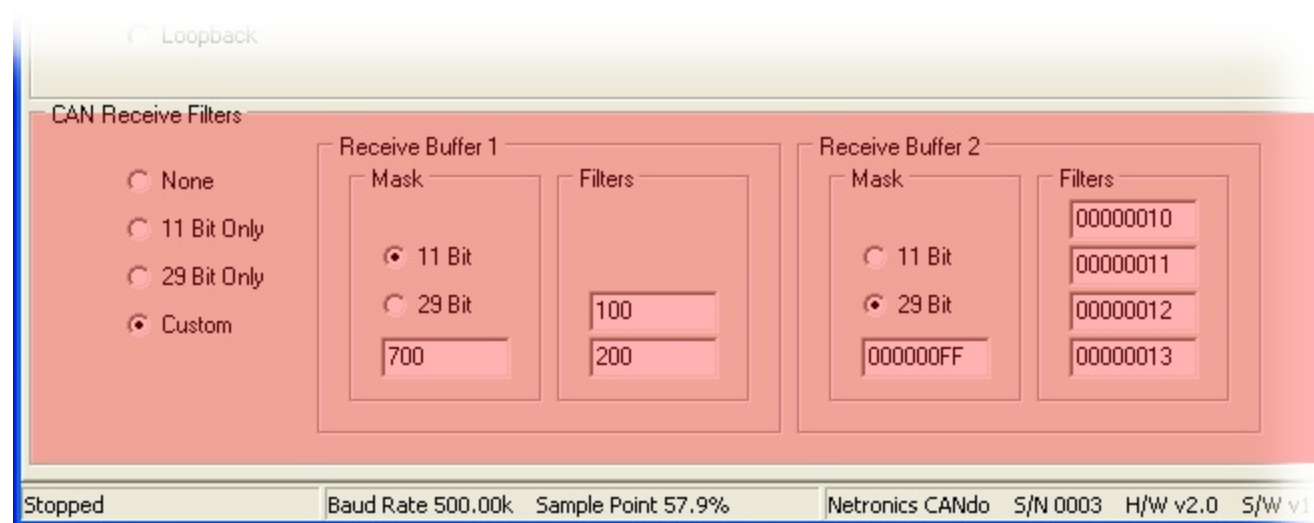
'Listen Only' mode, only allows messages to be received by CANdo. No low level acknowledgments are transmitted in this mode. This mode may be used to determine the baud rate of a CAN bus in a non-intrusive way. At least two other active CAN nodes must be present on the CAN bus for correct operation in this mode.

'Loopback' mode, is for testing purposes only & allows transmissions from CANdo to be received & checked without any other CAN nodes present. In this mode CANdo does not transmit on the CAN bus.

The operating mode cannot be changed while CANdo is running.

# CAN Receive Filters

**CANdo** contains two CAN message receive buffers that operate independently, collecting messages as they occur on the CAN bus. Normally these buffers are programmed to collect all messages & display them in the list on the 'CAN View' page. However, filters may be applied to the receive buffers to allow through only those messages with an ID that matches the filters. This is sometimes useful when looking for a particular message or when analysing a heavily loaded bus.



Each receive buffer contains a mask & a number of filters, two for 'Receive Buffer 1' & four for 'Receive Buffer 2'.

The mask identifies which bits within the ID are relevant for filtering. If the mask bit is set to a logic '1', then this bit is used to match the corresponding bit in the received message ID with each of the filters in turn. If there is a match between the relevant bits of the message & the relevant bits of at least one of the filters, then the message is accepted by the receive buffer.

Bits within the mask that are set to a logic '0' are ignored for filtering purposes. Hence, if the mask is all zeroes, '000' for an 11 bit mask or '00000000' for a 29 bit mask, then the receive buffer accepts all messages.

The receive filters cannot be changed while **CANdo** is running.

Using the example shown in the picture above, 'Receive Buffer 1' is configured to examine the top 3 bits of the ID of any 11 bit received message because the mask is set to 700 (hexadecimal), 11100000000 (binary). Hence, only messages with the top 3 bits of their ID matching at least one the filters are allowed through. The two filters are set to 100 (hexadecimal), 00100000000 (binary) & 200 (hexadecimal), 01000000000 (binary). Thus, the top 3 bits of the message ID must also be either 001 or 010 in order to be accepted.

11100000000 - Mask

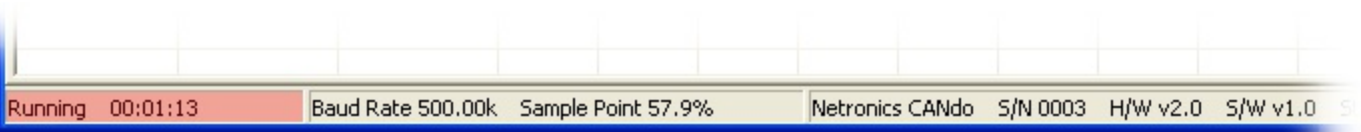
00100000000 - Filter

01000000000 - Filter

Hence, only 11 bit messages in the range 100 to 2FF (hexadecimal) are accepted.

# Run State

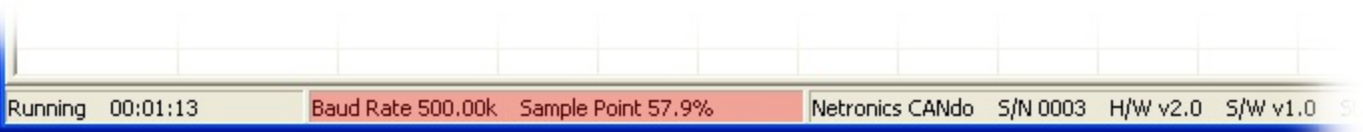
The first panel of the status bar displays the CANdo run state. While running, the elapsed time since starting CANdo is displayed in hours, minutes & seconds.





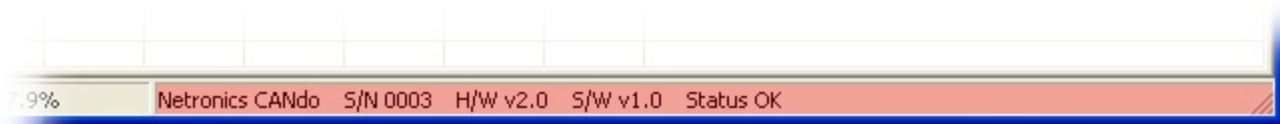
# Baud Rate

The second panel of the status bar displays the configured baud rate based on the settings on the 'CAN Setup' page.



# CANdo Status

The CANdo device status is displayed in the third panel of the status bar. The status is retrieved from the device on starting the application & then every second while CANdo is running.



The CANdo device status contains the following fields -

- Product description
- Serial number
- Hardware version
- Software version
- Status code

The status code displays the current internal state of the CANdo device. Normal operation is indicated by 'Status OK'. In the event of a detected fault, a code is displayed as 'Status ??', where ?? is an OR combination of one or more of the fault codes (in hexadecimal format) detailed in the table below.

Code	Fault
01	USB port receive message overrun
02	USB port receive message invalid
04	USB port receive message CRC error
08	CAN receive message no data
10	CAN receive message overrun
20	CAN receive message invalid
40	CAN transmit message overrun
80	CAN bus error



- **CANdo** is a compact USB to CAN interface that allows a PC to communicate directly with an embedded CAN bus
- **CANdo** is easy to install & use, deriving all necessary power from the USB port
- **CANdo** fully supports the CAN Specification Version 2, by Robert Bosch
- The **CANdo** firmware is flash programmable to allow for future upgrades
- An SDK is available to allow **CANdo** to be embedded in any Windows or Linux PC application written in a high level language

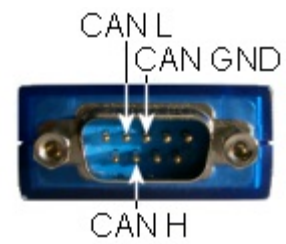
# LED Indicators

The LEDs on the topside of the **CANdo** housing indicate the status of the unit. The green LED indicates the system status & the yellow LED the CAN bus status.

LED State		Unit Status
Green LED	Off	No power to unit
	Flashing	System error
	On constantly	System OK
Yellow LED	Mainly off	High CAN bus load
	Flashing	CAN messages received/transmitted
	On constantly	No CAN messages received or transmitted

# Pinouts

The CAN bus connections are on the 9 way 'D' type connector in the **CANdo** housing.



Pin No.	Signal
1	N.C.
2	CAN L - CAN bus low
3	CAN GND - CAN bus ground
4	N.C.
5	N.C.
6	N.C.
7	CAN H - CAN bus high
8	N.C.
9	N.C.

N.C. - Not Connected

The CAN bus connections within the **CANdo** unit are not galvanically isolated from the USB connector.

# Specification

Parameter	Min.	Typ.	Max.	Units
<b>USB</b>				
USB supply voltage	4.0	-	5.25	V
USB supply current	-	45	80	mA
USB suspend current	-	330	500	uA
<b>CAN</b>				
Bus output voltage - dominant state				
CAN Bus Low	0.5	-	1.25	V
CAN Bus High	2.45	-	3.5	V
Bus output voltage - recessive state				
CAN Bus Low	-	2.3	-	V
CAN Bus High	-	2.3	-	V
Short circuit output current	-250	-	250	mA
<b>Environmental</b>				
Operating temperature range	-10	-	50	Deg. C
Storage temperature range	-40	-	85	Deg. C
<b>Mechanical</b>				
Enclosure dimensions	65mm (L) x 35mm (W) x 15mm (H)			
USB tethered cable length	1.0	1.1	1.2	m

# Certification

The **CANdo** unit is certified to comply with the essential requirements of the EU Directive 89/336/EEC, relating to EMC.

The **CANdo** unit specifically complies with the following standards.

- EN 55022 (class B, interference emission, trade and small industrial sector)
- EN 61000-6-2 (immunity, industrial interference immunity)
- EN 61000-4-2 (air discharge 8kV, contact discharge 4kV)
- EN 61000-4-3 (electromagnetic field compatibility 80MHz-1GHz, 10V/m)